

# Real-Time Traffic Separation in Shared Ethernet Networks: Simulation Analysis of the h-BEB Collision Resolution Algorithm<sup>1</sup>

Ricardo Moraes, Francisco Vasques  
Department of Mechanical Engineering  
University of Porto, Portugal  
e-mail: {rmoraes, vasques}@fe.up.pt

## Abstract

*In this paper, the enhanced collision resolution algorithm, referred as high priority Binary Exponential Backoff (h-BEB) is analyzed. Such algorithm provides high priority traffic separation, enabling the support of real-time communication in shared Ethernet networks. It allows multiple Ethernet standard devices to coexist in the same network segment with one h-BEB enhanced station, imposing higher priority for the transfer of h-BEB related traffic. The simulation analysis shows that the h-BEB algorithm guarantees values for both the average access delay and the related standard deviation, that are significantly smaller for the h-BEB station than for the BEB stations.*

## 1. Introduction

Despite of the full-duplex operating mode that has been introduced in the early 90s (IEEE 802.1D) [1], the vast majority of Ethernet networks still operate in heterogeneous environments, with Ethernet Switching Hubs interconnecting both independent node stations and Ethernet Repeater Hubs with multiple interconnected node stations (equivalent to shared Ethernet segments). In such heterogeneous environments, the Switching Hubs impose separate collision domains at each port (network segmentation). However, within each of the collision domains, the network still operates in the traditional shared Ethernet mode; that is, collisions are solved by means of a probabilistic contention resolution algorithm, *i.e.*, the medium access is inherently non-deterministic.

The major motivation for this work is to “propose and analyze solutions enabling the support of real-time communications in *shared* Ethernet environments, where Ethernet standard devices can coexist with modified/enhanced devices”. Basically, it is proposed the use of a modified algorithm for the collision resolution in shared Ethernet networks. The proposed

algorithm, referred as *high priority Binary Exponential Backoff* (h-BEB), provides high priority traffic separation, enabling the support of real-time communications in shared Ethernet networks.

In a previous paper [2], the probabilistic analysis of the h-BEB algorithm has been presented. Such analysis addressed the *exact analytical behavior* of the h-BEB algorithm, in heavily loaded network scenarios. However, it is well known that such type of exact probability analysis addresses a rarely occurring case, as it is based on the assumption that, at the start of the transmission attempt, all the network stations participate in the contention process. For more realistic load scenarios (intermediate load cases), the performance analysis must be done by simulation.

In a subsequent paper [3], the h-BEB collision resolution algorithm has been extended, allowing multiple h-BEB (real-time) stations to coexist in the same shared Ethernet segment and still imposing a higher priority for the transfer of privileged traffic.

## 2. The High Priority Binary Exponential Backoff Algorithm (h-BEB)

A station implementing the h-BEB algorithm has the same operating behavior as a BEB station, except for the backoff delay, which is set to 0. In such case, an h-BEB station starts immediately to transmit after the end of the jamming sequence. This behavior guarantees the highest transmitting probability to the h-BEB station, in a shared Ethernet segment with multiple BEB stations. The h-BEB station will always try to transmit its frame in the first available slot after the jamming sequence, while all the other stations implementing the BEB algorithm will wait between 0 and  $2^n-1$  slot times, where  $n$  is the number of collision resolution rounds. The h-BEB collision resolution algorithm is therefore able to impose real-time traffic separation, as the traffic generated by the h-BEB station will always be transferred before the traffic generated by the other

<sup>1</sup> This work has been partially supported by IDMEC (project AdvanSys) and by FCT (BD 13203/2003).

stations. Therefore, this algorithm is adequate to support real-time communications in shared Ethernet segments, as long as all the real-time traffic in the network is generated by the h-BEB station. The behavior of the h-BEB algorithm is completely described in [2].

### 3. The simulation model

A simulation model was implemented using the *Network Simulator (NS-2)* tool [4], considering a shared Ethernet environment, where multiple stations are interconnected with a *special station* implementing either the h-BEB (enhanced Ethernet mode) or the BEB algorithms (traditional Ethernet mode). The simulation model considers a 10 Mbps Ethernet network, where each station has a Poisson traffic source with a fixed packet length of 250 bytes. The total network load ranges from 40% to 110%. For each simulated load value,  $75 \times 10^4$  packets are successfully transmitted (that is, discarded packets are not considered).

The target of the simulations is to analyze the behavior of the h-BEB algorithm when compared to the use of the traditional BEB collision resolution algorithm. The performance measures include: throughput, average packet delay and standard deviation (transfer jitter). The throughput is the ratio between the total duration of the successfully transferred packets and the simulated time interval, *i.e.*, the fraction of nominal bandwidth that is used for successfully transferring data. The average packet delay is the average delay required to successfully transfer a packet, measured from the first transmission attempt to the end of the packet transfer. Discarded packets are not considered for the average packet delay evaluation, as this measure deals with just the successfully transferred packets. The standard deviation, which is related to the message transfer *jitter*, is given by:

$$\sigma = \sqrt{\frac{1}{N} \times \sum_{i=1}^N (x_i - \bar{x})^2} \quad (1)$$

where  $N$  is the total number of simulated packets,  $x_i$  is the delay of each transferred packet and  $\bar{x}$  is the evaluated average packet delay.

### 4. Performance Analysis

In this section, two simulation cases are analyzed: the *small population* case that considers the case of 5 interconnected stations with a network load ranging from 40% to 110% (the network load is equally shared by all the stations) and the *large population* case that extends the small population case to 65 interconnected

station. The BEB simulation results were validated against previous results presented by Christensen [5].

#### 4.1 The small population case

The enhanced Ethernet mode scenario, connects 1 *special station* implementing the h-BEB algorithm with 4 stations implementing the BEB algorithm, while in the traditional Ethernet mode scenario, all the 5 interconnected stations implement the BEB algorithm.

The average throughput for each Ethernet mode is plotted in Figure 1 and compared against the theoretical maximum throughput (*i.e.*, without packet collisions or packets being discarded). The achievable average throughput decreases in the 65% to 110% load region. The difference between the maximum throughput and the obtained throughput (both for the enhanced and the traditional Ethernet modes) is directly related to the lost packets. Similar results have already been shown in previous simulations made by other authors [5, 6]. Moreover, we observed that for the small population case, the h-BEB station never discards any packet. This is an expected and important result for a station implementing the h-BEB collision resolution algorithm.

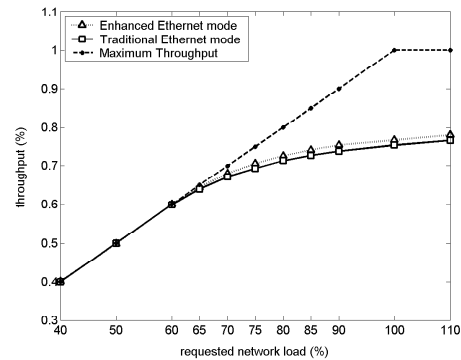
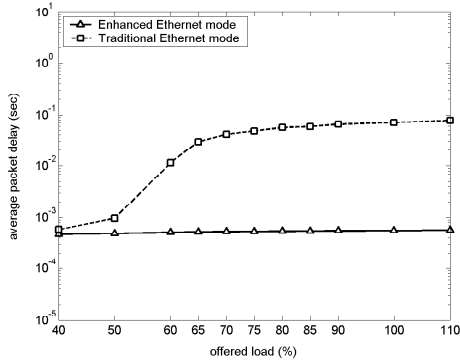


Figure 1: Throughput in a small population.

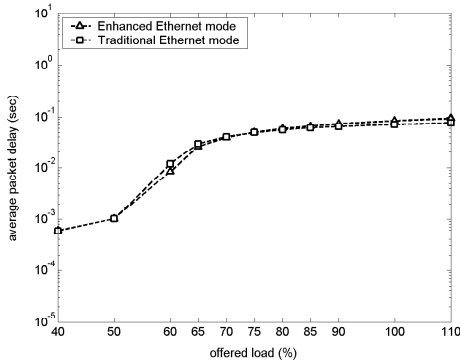
The average delay for transferring a packet in a small population Ethernet segment is represented in both Figures 2 and 3 for the case of, respectively, the special station, and all the other 4 stations. In both cases, the represented results compare the average delay for transferring a packet in the “enhanced mode” vs. the “traditional mode” scenarios. Figure 2 shows that for a network load above 50%, the h-BEB *special station* has an average packet delay much smaller than the BEB *special station*. For instance, for an 80% network load, the h-BEB station takes in average 0.547 ms to transfer a packet, while the BEB station takes in average 56.484 ms. More importantly, it is clear that, whatever the network load, the average packet delay is nearly constant for the enhanced network case scenario. This is a very important result, as it forecasts a predictable

communication delay when supporting real-time communications.

On the other side, Figure 3 shows that the *special station* has a very small influence on the behavior of the other 4 stations. That is, whatever the network case scenario (“enhanced” or “traditional”), the average packet delay of the BEB stations stays almost unchanged. It can be seen that there is a slight modification for network loads above 75%, due to the smaller number of occurring collisions.

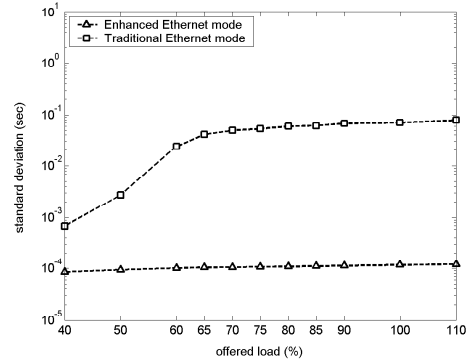


**Figure 2: Average delay for the special station in a small population Ethernet segment.**



**Figure 3: Average delay for the other stations in a small population Ethernet segment.**

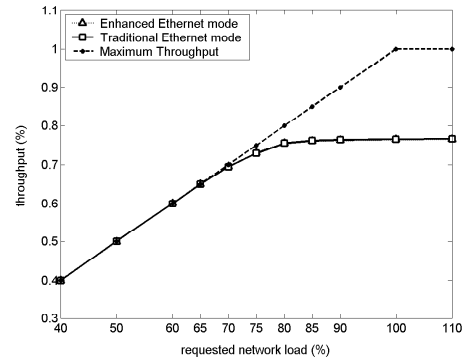
Finally, Figure 4 compares the standard deviation of the average packet delay, which is directly related to the message transfer *jitter*. From Figure 4, it becomes clear the difference between the “enhanced” and the “traditional” Ethernet mode scenarios. For the case of the *special station* implementing the h-BEB algorithm, the message transfer *jitter* is nearly constant and one order of magnitude smaller than the average delay, whatever the simulated network load. For all the other cases (stations implementing the BEB algorithm), the message transfer *jitter* is of the same order as the average packet delay, which is clearly not adequate for the support of real-time communications.



**Figure 4: Standard deviation for the special station in a small population Ethernet segment.**

## 4.2 The large population case

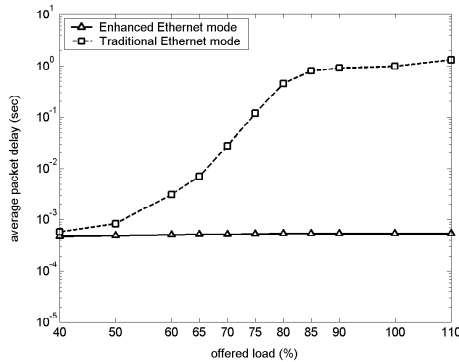
As in the small population case, the enhanced Ethernet mode scenario connects 1 h-BEB station with multiple BEB stations (64 stations), while in the traditional Ethernet mode scenario, all the interconnected stations implement the BEB algorithm.



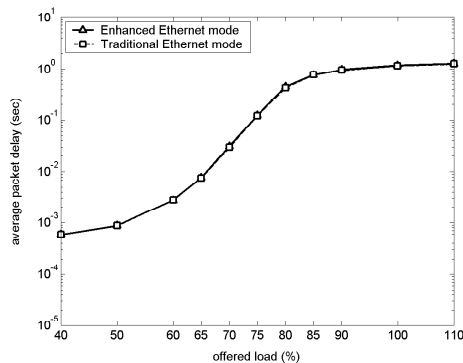
**Figure 5: Throughput in a large population.**

Figure 5 illustrates the average throughput for both the enhanced and the traditional Ethernet mode scenarios. It can be seen that the throughput is very similar in both scenarios, as the lines are almost superposed. The average throughput can also be compared with the theoretical maximum achievable throughput, which clearly shows that for network loads above 65%, there is a high rate of packets loss in the network (the difference between the maximum achievable and the obtained throughputs are directly related to the lost packets). As in the previous case, it has been observed that the h-BEB station never discards any packet, whatever the simulated network load. The average delay for the *special station* to transfer a packet in a large population Ethernet segment is represented in Figure 6. For the enhanced Ethernet mode, there is also a very small average packet delay. For example, for an 85% network load, the h-BEB station takes in average 0,536 ms to transfer a packet, while a BEB station takes

in average 800,13 ms. Similarly as for the small population scenario, in Figure 7, it was also observed that the *special station* has a very small influence on the behavior of the other 64 stations, as whatever the Ethernet mode scenario (“enhanced” or “traditional”), the average packet delay of the BEB stations stays almost unchanged (similar to results for small population).



**Figure 6: Average delay for the special station in a large population Ethernet segment.**



**Figure 7: Average delay for the other stations in a large population Ethernet segment.**

Both the results for the small and the large population scenarios clearly show that, whatever the network load or the network population, the average packet delay is nearly constant for the h-BEB station in the enhanced network case scenario. These are very important results, as they forecast predictable communication delays for the h-BEB station in very different load and population scenarios.

The difference between the “enhanced” and the “traditional” Ethernet mode scenarios in what concerns the standard deviation of the average packet delay were similar to those of the small population case.

## 5. Conclusion

This paper proposes the use of a modified algorithm: the h-BEB algorithm, for the collision resolution in shared Ethernet networks. In a previous paper [2], the

exact probabilistic analysis of the h-BEB algorithm has been presented, addressing a highly loaded network scenario. In this paper, more realistic network load scenarios have been analyzed by means of simulation. The performance measures included: throughput, average packet delay and standard deviation of the average packet delay (transfer jitter). Two cases were analyzed: a small population case and a large population case. In each of the analyzed cases, a comparison has been done between the results for the “enhanced” Ethernet mode scenario vs. the “traditional” Ethernet mode scenario.

For all cases it has been shown that the h-BEB collision resolution algorithm guarantees an average access delay significantly smaller for the h-BEB station. More significantly, *almost* constant values for both the average access delay and the related standard deviation have been observed for the traffic transferred by the h-BEB station. This is a very important result, as it forecasts a predictable communication delay when supporting real-time communications.

## 6. References

- [1] "IEEE standards for local and metropolitan area networks: virtual bridged local area networks," in *IEEE Std 802.1Q-1998*, 1999.
- [2] R. Moraes and F. Vasques, "A Probabilistic Analysis of Traffic Separation in Shared Ethernet Systems Using the h-BEB Collision Resolution Algorithm," presented at 13th International Conference on Real-Time Systems - RTS'2005, Paris - France, 2005.
- [3] F. Carreiro, R. Moraes, J. A. Fonseca, and F. Vasques, "Real-Time Communication in Unconstrained Shared Ethernet Networks: The Virtual Token-Passing Approach," submitted at *Emerging Technologies and Factory Automation - ETFA*, Catania, Italy, 2005.
- [4] "ns-2 Network Simulator," 2.27 ed, 2004. Available at: <http://www.isi.edu/nsnam/ns/>
- [5] K. J. Christensen, "A simulation study of enhanced arbitration methods for improving Ethernet performance," *Computer Communications*, vol. 21, pp. 24-36, 1998.
- [6] D. R. Boggs, J. C. Mogul, and C. A. Kent, "Measured capacity of an ethernet: myths and reality," *Computer Communication Review*, vol. 25, pp. 123-136, 1995.